(19) Canadian Intellectual Property Office
An Agency of Industry Canada

Office de la Propri,t, Intellectuelle du Canada
Un organisme d'Industrie Canada

(11) CA 2 493 897 (13) A1

(40) 22.04.2005
(43) 22.04.2005

(12)

(21) 2 493 897

(51) Int. Cl.7: H04L 9/32, H04L 12/16

(22) 24.01.2005

(30) 60/579,890 US 16.06.2004
60/605,150 US 30.08.2004

(71) SXIP NETWORKS INC.,
206 - 55 Water Street, VANCOUVER, B1 (CA).

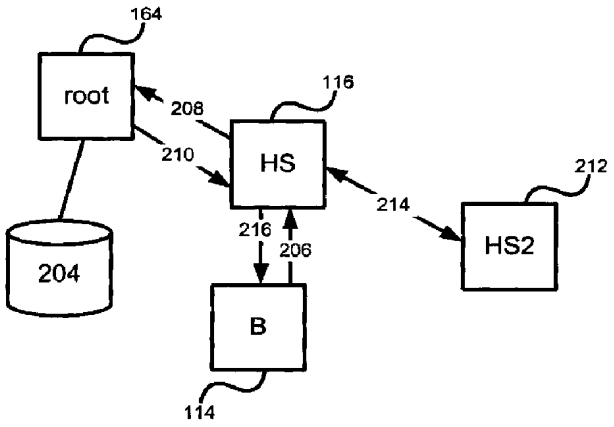
(72) HARDT, DICK C. (CA).

(74) BORDEN LADNER GERVAIS LLP

(54) GESTION REPARTIE D'INFORMATION SUR LES CONTACTS
(54) DISTRIBUTED CONTACT INFORMATION MANAGEMENT

(57)

A method and system for interaction with webservices and for performing distributed contact management use standard interfaces to communicate with other entities in an identity management network. The use of homesites as user data stores allows for homesite to homesite communication to allow for distributed contact management, and the generic interface allows for homesite to webservice interaction.



Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2493897 A1 2005/04/22

(21) 2 493 897

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(22) Date de dépôt/Filing Date: 2005/01/24

(41) Mise à la disp. pub./Open to Public Insp.: 2005/04/22

(30) Priorités/Priorities: 2004/06/16 (60/579,890) US; 2004/08/30 (60/605,150) US

(51) Cl.Int.^{7}/Int.Cl.^{7} H04L 9/32, H04L 12/16

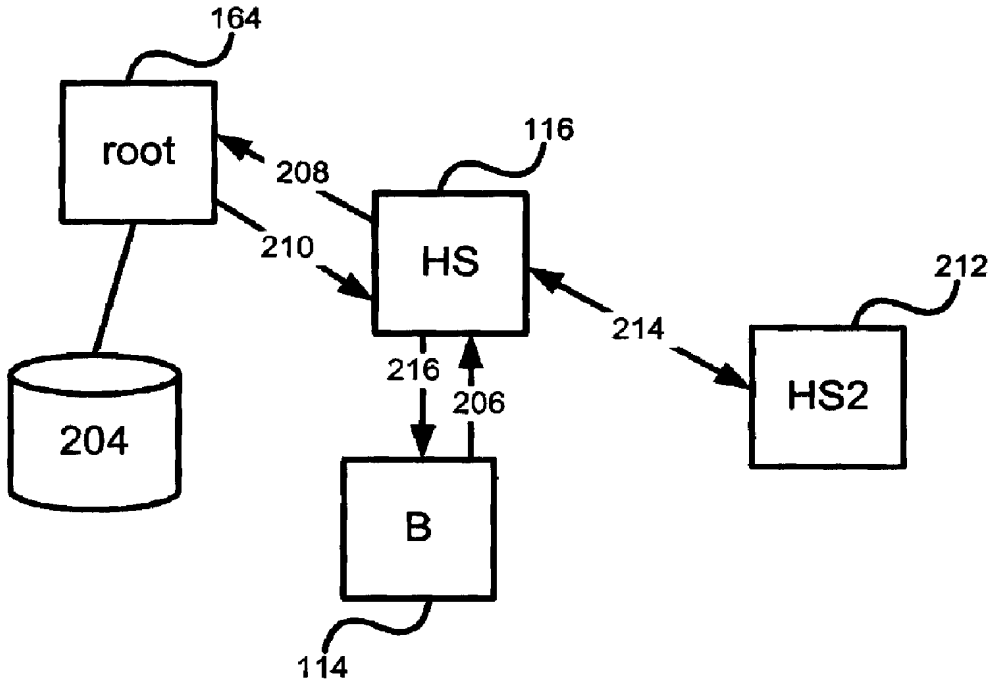
(71) Demandeur/Applicant:
SXIP NETWORKS INC., CA

(72) Inventeur/Inventor:
HARDT, DICK C., CA

(74) Agent: BORDEN LADNER GERVAIS LLP

(54) Titre : GESTION REPARTIE D'INFORMATION SUR LES CONTACTS

(54) Title: DISTRIBUTED CONTACT INFORMATION MANAGEMENT



(57) Abrégé/Abstract:
A method and system for interaction with webservices and for performing distributed contact management use standard interfaces to communicate with other entities in an identity management network. The use of homesites as user data stores allows for homesite to homesite communication to allow for distributed contact management, and the generic interface allows for homesite to webservice interaction.

ABSTRACT

A method and system for interaction with webservices and for performing distributed contact management use standard interfaces to communicate with other entities in an identity management network. The use of homesites as user data stores allows for homesite to homesite communication to allow for distributed contact management, and the generic interface allows for homesite to webservice interaction.

DISTRIBUTED CONTACT INFORMATION MANAGEMENT

FIELD OF THE INVENTION

The present invention relates generally to electronic identity management systems. More particularly, the present invention relates to authentication and security for data exchange in a distributed hierarchical identity management system.

BACKGROUND OF THE INVENTION

In the field of identity management, there are a number of known systems for providing user identity services on the Internet. Microsoft's Passport ™, and the Liberty Alliance identity management system are two such known examples, as are the identity management systems taught in Canadian Patent No. 2,431,311, and Canadian Patent Application Nos. 2,458,257, 2,468,351, and 2,468,585.

Many known identity management systems offer secure logins, allowing a user to visit a site in the network (membersite) and obtain a secure login to that site using an identity store to authenticate the user identity over a secure channel. The use of a secure channel allows an identity store to provide the membersite with user login information and/or confidential user information.

However, the reliance on secure channels increases the barrier to entry for membersites. Under a secure setup, lightweight, or simple, login is encumbered by the overhead of a secure channel.

In an identity management system that relies upon homesites to act as an identity store which stores user identity information, it may be advantageous to provide a form of graduated security to allow a membersite to obtain identity information, including authentication, using a number of different channels, each with different security features.

There is a further need for a mechanism through which a webservice provider can obtain user authentication and authorization for a third party to receive information. At present, if a third party wishes to aggregate information from a number of webservice providers for the user, or if a third party requires information from a webservice provider to further process before providing the results to a user, the third party and the webservice must

be heavily linked. Typically, the third party must become associated with the webservice, and have its services bundled by the webservice provider. Thus a financial institution can use an aggregation service to perform analysis on a client's holdings, but a client cannot easily obtain an aggregation across a number of financial institutions. There is therefore a need for a mechanism for third parties to provide authentication of a user authorization for release of information provided by a webservice.

There are at present a number of contact management services that allow a user to provide a list of known contacts. If the contacts provided a user subscribe to the same service, when one of the users updates a segment of a profile, the change is automatically reflected in the other users contact list. However, at present, these services are highly centralized. There is no automated mechanism to obtain information about users that have not subscribed. There is a plurality of these services, and at present there is no convenient mechanism for data exchange between them. This results in users forming small collective islands of contact sharing. There is a need for a distributed contact management system that allows users to share information with people in a vast identity management system that allows for automated updating of contact information.

It is, therefore, desirable to provide an identity management system that can provide at least one of improved gradations in the security levels, support for third party webservices and support for distributed contact management.

SUMMARY OF THE INVENTION

It is an object of the present invention to obviate or mitigate at least one disadvantage of previous identity management systems.

In a first aspect of the present invention, there is provided a method of obtaining contact information from a node in an identity management network. The method comprises receiving a homesite identifier and a user identifier associated with an email address; requesting from a homesite associated with the received homesite identifier contact information for a user associated with the received user identifier; and receiving from the homesite associated with the received homesite identifier the requested contact information.

In an embodiment of the first aspect of the present invention, the method further includes the step of forwarding the received contact information to a user. In another

embodiment, the step of receiving a homesite identifier includes receiving the homesite identifier and user identifier from a network root and is optionally preceded by the step of transmitting a request for a user identifier and a homesite identifier associated with an email address. In these embodiments, the user identifier can be a globally unique persona identifier. In another embodiment, the step of receiving the requested contact information is preceded by obtaining approval for the release of the contact information from the user associated with the requested contact information. In another embodiment, the step of receiving the requested contact information includes receiving a universal resource indicator for receiving updated contact information.

In a second aspect of the present invention, there is provided a method of authorizing a webservice provider to release information, associated with a user, to a third party. The method comprises the steps of receiving from the webservice provider a request for user authentication and authorization to release information to the third party; authenticating the user and obtaining user authorization for the release of the information; and forwarding to the webservice provider authorization from the authenticated user to release the information to the third party.

In an embodiment of the present invention, the received request from the webservice provider is received via the third party. In another embodiment, the received request includes an explanation of the information to be released to the third party, the explanation preferably includes a text explanation to be shown to the user prior to obtaining user authorization and a programmatic explanation. In another embodiment, the received request includes a nonce and the step of forwarding includes forwarding the nonce along with the response. In another embodiment, the step of forwarding includes signing the response, appending proof of authoritativeness for the user and forwarding the authorization to the webservice provide via the third party.

Other aspects and features of the present invention will become apparent to those ordinarily skilled in the art upon review of the following description of specific embodiments of the invention in conjunction with the accompanying figures.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the present invention will now be described, by way of example only, with reference to the attached Figures, wherein:

Fig. 1 is a flowchart illustrating a method of the present invention;

Fig. 2 is a flowchart illustrating a method of the present invention;

Fig. 3 is a block diagram illustrating a lightweight login in a system of the present invention;

Fig. 4 is a block diagram illustrating an increased security login in a system of the present invention;

Fig. 5 is a block diagram illustrating a further increased security login in a system of the present invention;

Fig. 6 is a block diagram illustrating a secure login in a system of the present invention;

Fig. 7 is a flowchart illustrating a method of the present invention;

Fig. 8 is a flowchart illustrating a method of the present invention;

Fig. 9 is a block diagram illustrating the dataflow for providing authorization and authentication to a webservice provider;

Fig. 10 is a block diagram illustrating an additional dataflow for authorization of a webservice provider;

Fig. 11 is a block diagram illustrating a rich client interfacing with a homesite and a webservice provider;

Fig. 12 is a block diagram illustrating a distributed contact management system of the present invention;

Fig. 13 is a block diagram illustrating an automated user information distribution system of the present invention; and

Fig. 14 is a block diagram illustrating a further embodiment of the automated user information distribution system illustrated in Figure 13.

DETAILED DESCRIPTION

Generally, the present invention provides a method and system for identity management supporting graduated security levels, third party web services and contact management.

In the following discussion, a hierarchical distributed identity management system is assumed, though one skilled in the art will appreciate that a number of these techniques can be utilized in other identity management networks or other environments where transactional security is of importance.

In view of the need for a graduated security mechanism, the system of the present invention can provide a series of different levels of security. As noted with regard to the prior art, a mechanism for providing a series of graduated security levels provides membersites with the ability to determine the degree of security that they require. For sites that relied upon identification by the presence of a cookie, as many news based website do, or sites that relied upon simple username password combinations transmitted in cleartext prior to joining the identity management network, there is little need for requiring a very secure user identification channel. For such sites, requiring a secure login with signature verification for exchanged data serves as a barrier for entry. On the other side of the equation, financial service websites or websites providing medical histories are best served by very secure logins, with a mechanism that reduces the ability of malicious parties to perform man-in-the-middle type attacks. To provide such a varied login, the system of the present invention allows for a graduated security login. The graduated security login can use both differentiated levels of user authentication and differentiated levels of channel security.

In prior art identity management systems, users are authenticated by providing a user identifier, such as a username, and a shared secret, such as a password. In other systems, typically reserved for specialty uses, other information was used in place of a shared secret, including fingerprint or biometric data. If the provided information was sufficiently unique, as it is with fingerprint and biometric data, the provision of this information was sufficient, and a user identifier was not required. Thus, depending on the level of security required for the authentication, different information has been required. However, in the prior art, login information has been globally set, so that regardless of what a user may want to do the same

authentication test was applied. In the context of a membersite requesting user authentication, this is particularly cumbersome. A user, who is being authenticated by a news site so that a particular presentation layout can be selected based on user preference, does not need to be bothered with a request to authenticate using a user name and password, especially if the user has been recently authenticated. Additionally, the news site does not necessarily want the user to be authenticated by a username and password combination, or an even stronger authentication mechanism, as it makes the process too cumbersome for the user and diminishes the likelihood that the user will visit the site. Conversely, a financial institution may want the user to be authenticated by a homesite regardless of when the user was last authenticated, and may demand that that authentication. Similarly, a medical database may want to force the user to authenticate with the homesite and use a particularly robust authentication mechanism, such as a biometric scan, so that there is confidence that the user has been properly authenticated.

To service the varied needs of different membersites, the homesite can support a series of different authentication levels. By supporting the plurality of different authentication mechanisms, the homesite can receive requests from a membersite to authenticate at a certain level of security. Additionally, a user can set a preference that when certain information, such as a credit card number, is requested, the homesite will only release it if authentication at a predefined level has been obtained. Thus, when a homesite receives a request for user information or user authentication, it can determine from both the request, and the requested information, the level of user authentication that is required. If the request for authentication and the requested information specify different levels of security, the homesite can use the higher of the two for the maximum security.

One skilled in the art will appreciate that different dimensions of security can be applied independently of each other. Differing levels of user authentication security can be applied, so that users can be required to provide different complexities of authentication information, as can differing levels of security in the communication channel formed between the membersite and homesite through the user's browser. As a further dimension to the security level, a time sensitivity factor can be required of the user authentication, so that differing levels of user authentication security can be combined with a staleness factor that allows authentication of a user within a fixed period of time to be varied.

Figure 1 illustrates an exemplary embodiment of a method for executing the above-described graduated authentication system. In step **100** a homesite receives a user identity request from a membersite. A user identity request typically includes a request for one or both of user authentication and information about the user. In step **100** the homesite determines a required security level in accordance with the request. This determination can take the form of examining the request to determine a membersite's explicit request for a security level, examining user preferences which indicate that a user wishes to authenticate with nothing less than a specified authentication security level, examining a user specified minimum authentication security level associated with requested identity information, and optionally a set of homesite policies regarding the authentication of users at minimum security levels can also be used in the determination of the required authentication security level. The determined security level is preferably a combination of both authentication security and a time limit. The time limit defines an acceptable level of staleness in the authentication, allowing the combined security level either to force the user to authenticate, or to allow a previous authentication. In step **104**, the user is authenticated using an authentication mechanism having combination of a security level and time factor at least equal to the determined combined security level. For example, the determined combined security level could be that the user has been previously authenticated using a username and password, has been authenticated within a predetermined time limit previously, the user must authenticate for this transaction using a user name and password pair, a username and password pair in conjunction with another shared secret, or with biometric authentication. The previous list is intended to be exemplary, and in no way should it be considered exhaustive of either of the two dimensions of security or of their combination. Upon successful user authentication, the homesite provides the membersite with a user identity response in step **106**. This response preferably contains the requested user information, and a statement from the homesite detailing the security level at which the user was authenticated. One skilled in the art will appreciate that the membersite can include in the user identity request a list of acceptable authentication mechanisms, and the homesite would them determine the required security level by selecting one from the provided list, optionally doing so in accordance with user preferences and homesite policies.

Thus, a homesite can use any of a number of authentication methods, and preferably uses the one specified by the membersite. To allow for authentication methods to be properly specified, each authentication method can be assigned a security level, allowing the membersite to request authentication at a desired level. The homesite can then use any authentication method at, or above, that level to authenticate the user.

From the perspective of a membersite, when a user visits, the membersite can determine that the user has a homesite, through any number of known mechanisms, including looking for a cookie in a shadow domain. The user can be redirected to the homesite with a request for authentication, possibly including an information request. This request includes an indication of the security level, preferably for both the authentication and the time sensitivity, required. When a response from the homesite is received, the response can include a statement, preferably signed by the homesite, that authentication was performed at a given level either at or in excess of the one specified by the membersite.

Figure 2 is a flowchart illustrating an exemplary embodiment of a method to implement the above-described graduated authentication method from the perspective of the membersite. In step **108**, the membersite issues a user identity request containing a defined authentication security level. In step **110**, the membersite receives the homesite's user identity response. In step **112**, the membersite examines the user identity response to determine the security level at which the user was authenticated. The determined security level can then be compared to the security level defined in step **108**. If the level does not meet the requirements, the membersite can handle the error in any number of ways. As an example, if a membersite is a financial institution that in a login procedure obtains the user login information from a homesite, and the membersite specifies that the user must authenticate using a user name and password combination, and receives an identity response that was authenticated on the basis of a previous authentication, the membersite can refuse the user login. The membersite can then present the user with a notice that the homesite did not use the required authentication and then query the user for account information for an out-of-identity-management-network login.

As an example of the above described authentication security levels, consider the scenario of a news server that provides users with specified layout and content filtering based on saved user profiles. When a user visits the news server, the server sends the user

to the homesite for authentication, and specifies that the lowest form of authentication is required, which in this scenario is that the user possesses a cookie from the homesite indicating that an authentication has occurred in the last 30 days. The homesite receives the user authentication request, determines that the user identifier, such as a globally unique persona identifier or a pairwise unique identifier, can be released without obtaining further user authentication as the user has previously authenticated. The user's identifier, along with a statement that authentication has been performed at or above the desired level, is then provided to the news server in a response signed by the homesite. The news server can then cross reference the user identifier with a set of preferences to display the news content in the desired format. Upon reading a story, the user clicks on a link to purchase a photo associated with one of the news stories. The purchase will be done on a credit card, whose information is stored by the homesite. The news server sends a request for user information to the homesite and requests the user's credit card number and a shipping address. The news server requests that the homesite authenticate the user using at least a username and password combination. The homesite receives the request for user information, and checks the user preferences related to the release of information. These preferences indicate that though the user will release a shipping address from a username and password challenge, a stronger challenge, such as a username and a response to two personal identification questions selected from a pool of questions, must be used to release a credit card number. The homesite then randomly selects two questions from a pool of questions, including information such as birthdate, place of birth, mother's maiden name, a favorite color, and a pet's name. These questions are provided to the user as a challenge. Upon successful completion of the challenge, the information is released to the news server in a signed response that includes an indication that a challenge at least as rigorous as the username password was obtained. Other levels of security can include a biometric or fingerprint scan, an out of band challenge such as a telephone call placed to a designated phone number, an out of band challenge including a password request in the out of band connection, automated token generation systems, and other known authentication mechanisms. One skilled in the art will appreciate that the above list is intended to be exemplary and not limiting in any manner.

As a companion to the above authentication security levels, the present invention can optionally provide a series of different channel configurations so that the channel between the membersite and homesite can have different levels of security itself. These two systems can be implemented independently of each other, though in combination they provide a large number of security options.

Figure 3 illustrates the data flow between the browser B **114** of a user, whose identity information is stored by homesite HS **116**, when attempting a login to membersite MS **118**. B **114** connects to MS **118** over a data connection **120**. The degree of security required for the authentication operation can be determined by the needs of the membersite. In transactions that do not require high degrees of security, such as authentications that would otherwise be username and password pairs exchanged in the clear, encryption is not required at either end. As a result, after browser **114** connects to MS **118** over datapath **120**, MS **118** requests authentication of the user by sending an authentication request to HS **116**. This request is sent to HS **116** by sending an authentication request to B **114** over datapath **112**. The request sent to B **114** contains a redirect command that redirects B **114** to HS **116** and sends the authentication request over datapath **126**. Thus, the authentication request is sent over a virtual channel created by datapaths **122** and **126** connected by the user redirection shown as **124**. HS **116** authenticates the user, using any of a number of techniques as described above, or in the prior art references, and then provides the requested information to MS **118** by sending it, via browser **114**, on over the channel created by datapaths **128**, **130** and **132**. If MS **118** would have otherwise used a simple username and password pair transmitted in the clear, the authentication of the user at HS **116** may be done over a secure channel, but the data provided to MS **118** can be send in the clear over unsecured data path **132**. This allows sites that do not require secure connections to belong to the identity management network without supporting secure connections. In a presently preferred embodiment, HS **116** will send a response to MS **118** using the same data connection type that MS **118** sends the authentication request using, unless otherwise specified. Thus, upon receiving the authentication request over unsecured channel **124**, HS **116** provides the requested authentication to MS **118** over unsecured channel **130**.

Figure 4 illustrates the dataflow for a scenario where MS **118** requests data over an insecure channel and HS **116** is required to send the data over a secure data channel. When

making the authentication request, MS **118** may consider that though the confirmation of the user identity is confidential, the request for the information is not. As such, MS **118** may choose to not use a secure channel to request the authentication. MS **118** then transmits an authentication request to HS **116** over the unsecure virtual channel created by datapaths **122**, **124** and **126**. The request specifies that the response should be transmitted over a secure channel. This allows MS **118** to not cause a redirect to its own secure server at the time of making the request, and instead simply sends the user to HS **116**. After authenticating the user of browser **114**, and optionally obtaining user authorization, HS **118** redirects the user to MS **118** over secure channel **136**. Security for the channel can be provided in any of a number of ways including the use of Secure Sockets Layer (SSL) connections, or the secure hypertext transfer protocol (https). If MS **118** requests more than authentication, and includes a request for user information, such as biographic or financial data, the secure return path **136** provides security for the transmitted data. One skilled in the art will appreciate that if a user specifies that certain data is only to be released over secure channels, HS **116** can, in response to a request, redirect browser **114** to MS **118** to provide the message that the response can only be provided over a secure link. Thus, the user can be guaranteed that confidential information is only provided in secure sessions.

In certain attacks on secure servers a "man in the middle" is used, so that requests for information are intercepted, modified, and then passed along. If a man in the middle type attack of this sort is attempted on the system of Figure 4, HS **116** will receive a request for additional information, but will only send it over a secure channel. Nonetheless, it may be beneficial for MS **118** to be able to easily identify such attacks. To allow for this, a further gradient of security can be introduced. Such a further security gradient is illustrated in the dataflows of Figure 5. After B **114** connects to MS **118** over connection **120**, MS **118** makes an authentication or information request from HS **116**, by redirecting B **114** to HS **116** over the virtual channel created by datapaths **122, 124** and **126**. After authenticating the user of B **114**, and optionally obtaining user approval for the release of information, HS **116** sends the requested information or authentication to MS **118** via B **114**, by redirecting B **114** over the virtual channel created by datapaths **140 142** and **144**. However, in addition to using a secure channel, HS **116** includes in the response the parameters of the request. MS **118**,

upon receipt of the response, can then easily identify if the parameters have been modified. This alerts MS **118** to the start of a man in the middle attack.

Figure 6 illustrates a further security level for use in the present invention. The user of browser **114** visits MS **118**, and, upon indicating a membership in the identity management network, is redirected to HS **116** along the virtual channel created by datapaths **146**, **148** and **150**. After authenticating the user, HS **116** transmits the response to the information and/or authentication request to MS **118** over secure the virtual channel created by datapaths **140** **142** and **144** along with the request parameters. To provide enhanced data protection, MS **118** uses secure paths **146** and **150** to transmit the request to HS **116** and also signs the request. HS **116** can then verify that the request was signed by MS **118**, and has not been tampered with during transmission. If a request has been tampered with, HS **116** can redirect B **114** to MS **118** without the requested information to provide a message that the request was modified prior to receipt. If HS **116** and MS **118** have no other connection to each other, other than belonging to the same identity management network, MS **116** can provide its public key to HS **116** along with the request. To ensure that the signature is not modified, or replaced, during an attack, the signature can be signed by a common trusted party, such as a network root or a trusted certificate authority.

To offer the different gradients of channel security, the present invention provides for both membersites and homesites to communicate to each other, preferably through browser **114**, using a channel selected from a channel listing. The following listing is meant to be exemplary and is not necessarily exhaustive. The list is not strictly ordered to show increasing security, as certain features of some channels offer security in a different manner than others. At a first level an open channel, with no encryption, can be used between the MS and the HS. To increase the security, and open channel can be used with HS signing the response to show that the content has not been modified in transit. A secure channel can be used, so that transit between the HS and B, and B and MS is secure. A secure channel with a signed response allows HS to have a secure connection to B, and then have a secure channel from B to MS, and allows MS to see that the response has not been modified in transit. An open channel can be used, with both the request and the response signed. This allows HS to know that the request for information has not been tampered with, and allows the MS to know that the response has not been tampered with. If HS passes the signed

request back to MS along with the signed response, MS can also verify that the request was not tampered with. The same signed request and response can also be transmitted over a secure channel.

By offering a series of these security levels the identity management system of the present invention allows membersites to use the most appropriate security for their needs, and does not force a one size fits all solution upon membersites. Homesites include input ports to receive requests for information and authentication. Prior to release of the information or authentication, a homesite can examine the information to be released and compare it to specified user conditions for the release of that information. Thus, a user can specify a channel security level at which information can be released, similar to the authentication security level settings on information described above. This allows a membersite to make a low security request, and a user preference or homesite policy to override it, and inform the membersite that the requested information can only be released using secure channels. The use of redirect commands allows the HS and MS to pass these messages to each other transparently to the user. Thus, the homesite input ports receive membersite requests, while an authentication engine obtains user authentication, and optionally obtains user authorization for the release of requested information. A homesite response engine then prepares the response to the received request and transmits it to the membersite over either the requested channel, or over a channel required by user preferences or homesite policies.

MS **118** is always guaranteed that the message from HS **116** has not been modified when a signed response is sent. The signature can be verified against a signature signed by a trusted third party, such as a network root as described in other references, or by a certificate authority.

Figure 7 illustrates an exemplary method for a homesite to select a channel as described above. In step **152**, a homesite receives a user identity request. In step **154**, the homesite determines a required channel security level in accordance with the request. As described above the channel can be selected from a list defined a priori. In step **156**, the homesite provides a user identity response over a channel having the appropriate security level. As one skilled in the art will appreciate, the determination of a required channel can be done in accordance with both the request and user or homesite defined preferences. As

described above, if a membersite requests information over a channel deemed unacceptably low by either homesite or user defined preferences, the homesite can attempt to force the membersite to use a higher security channel by redirecting the user to the membersite with a response that indicates that a more secure channel is required. This response can either simply indicate that a more secure channel is required or it can indicate the minimum channel required.

Figure 8 illustrates an exemplary method for a membersite to specify a channel as described above. The membersite, in step **158**, issues a user identity request, which includes a defined channel security level. As described above the channel security level can be selected from a list defined a priori. In response to the user identity request, the membersite receives, in step **160**, a user identity response. In step **162**, the membersite can examine both the response and the channel, over which the response was received, to determine that the channel has the defined security level. If the membersite requested that the response include a signed set of the request parameters over an unencrypted http channel, an inspection of both the channel and the response will indicate whether or not the response meets the requirements. If the response does not meet the requirements, the user can be informed that the homesite did not respond as expected and that an attack may be in progress on the user's identity. In the alternate the membersite could determine that it is under attack from a malicious third party, and determine that the safest course of action is to terminate the connection and log the IP address of the response sender, which, if the browser was used to redirect the response, should correspond to the user.

One skilled in the art will appreciate that the above described channel and authentication security levels can be provided either separately from each other or in tandem. They both rely upon a membersite issuing a user identity request with a defined security level, and the membersite receiving a response that is checked to ensure that it meets the defined security level. From the perspective of the homesite, both method involve receiving user identity requests, determining a security level in accordance with the received request and sending a response that meets the specified security levels.

Figure 9 illustrates the application of the security system described above to allow a membersite to connect to a webservice on behalf of a user to obtain information or to have service performed. In Figure 9, HS **116**, MS **118** and webservice WS **166** all belong to the

identity management network operated by root **164**. In belonging to the network, each node has trust in root **164**, and can identify other nodes in the network by requesting a signature associated with the other nodes that is signed by the root. By offering another node in the network a public signature block that is signed by the root, a node can establish both that it is part of the network, and that any transmission that it makes has not been tampered with.

When the user of browser **114** establishes a session with MS **118**, over connection **168**, an authentication with HS **116** takes place (not shown as part of the data flow). After authentication, the user may request a feature provided by MS **118** that requires access to a webservice, such as WS **166**. As an illustrative example, not intended to limit the scope of the invention, MS **118** may offer a financial portal service to the user of browser **114**, whereby MS **118** collects financial information from a number of other servers and presents it to the user in a consolidated format. WS **166** can match a globally unique persona identifier (GUPI) with the user of browser **114** and the services to which the user is subscribed. MS **118** provides a request to WS **166** over datapath **170**. WS **166** sends a request **172** to MS **118**. This request typically includes a request for user authentication and a set of information to allow WS **166** to identify the user. The request from WS **166** can also include requests for assertions from third parties that are held by the homesite **116**. Such assertions can include verifiable statements that a user is a member of an organization, such as a reward program, and even that the user has obtained a status level in the organization. Other assertions may be issued by governmental organizations indicating that a user has a geographical location. Those skilled in the art will appreciate that any number of third party assertions can be provided to WS **166**. This request is preferably accompanied with a nonce or other form of session identifier so that MS **118**, or another system, is prevented from using the user authentication as part of a replay attack. MS **118** forwards the request for authentication to HS **116** by redirecting the user along logical datapath **174**, one skilled in the art will appreciate that datapath **174** can include multiple channels established between different nodes on a point-to-point basis. The request from MS **118** to HS **116** may simply be the WS **166** request, or it may include a series of requests, including an aggregation of requests from the number of other web services (not shown). Furthermore, the request sent along datapath **174** may include other information needed by MS **118**. The request relayed to HS **116** preferably contains a request for a set of information about the user, user authentication, and

an explanation of what information is being provided and why it is being requested. In a presently preferred embodiment, the explanation is provided both as plaintext so that HS **116** can easily display it to the user, and as a programmatic explanation, so that HS **116** can obtain one-time authorization for the release of the information to WS **166**. The programmatic explanation, if provided, allows HS **116** to simply perform a compare operation on existing authorizations, reducing the number of times that the user must interact with HS **116**, increasing the appearance of a seamless experience.

Upon obtaining user authorization and authentication, HS **116** prepares a response, signs the response and includes its public signature, signed by root **164**. If the request from MS 118 is an aggregation of requests from multiple webservices, HS **116** can sign each corresponding response separately so that each webservice is provided with only the information that it requested. In an alternate embodiment, to reduce the computational overhead on HS **116** imposed by signing multiple data blocks, the entire response is signed, and each web service is provided the whole response. The response is sent to MS **118** via browser **114** over datapath **176**. MS **118** preferably breaks the response into the separately signed segments and forwards each segment to the respective WS. WS **166** then receives its request on datapath **178**. WS **166** can then authenticate that the data has not been modified in transit by examining the homesite signature and knowing the root signature. The use of a nonce, as described above, provides WS **166** the ability to track when the request was issued if a timeout value is to be applied. WS **166** can match information in the response from HS **116** to information held, such as bank account information, to determine which information to release to MS **118**. Upon validating the authorization and gathering the information to release to MS **118**, WS **166** sends the information to MS **118** over datapath **180**. Upon receipt of the information from WS **166**, MS **118** can act on the information as required. Depending on the content of the response, WS **166** may select the elements of the signed response that it needs and then examine the authorization it has received. If authorization has been received WS **166** will either provide a token to MS **118** that permits multiple access without further authentication, or will provide the requested information to MS **118** without a token to provide one-time access only. For the purposes of an example, not intended to limit the scope of the present invention, the following scenario is presented. A user directs browser **114** to MS **118**, where a session has already been established. MS **118**

provides the ability to aggregate information, such as travel information, for a user. MS **118** has knowledge of the user's upcoming travel itinerary, and proceeds to connect to an airline travel webservice, WS **166**. WS **166** upon receiving the initial contact from MS **118** provides a request for authentication of the user, using datapath **172**, and requests the user's full name, address and frequent flier information. This request is forwarded to HS **116**, possibly along with other information requests, following datapath **174** through MS **118**, and browser **114**. Upon receipt of the request, HS **116** requests that the user re-authenticate. The request from WS **166** is accompanied by both a text explanation outlining the information that is going to be released and a programmatic explanation; so that at a later date the user does not need to interact with HS **116**, and HS **116** can simply send the response. After authentication and acceptance of the release of the information, the user authorizes HS **116** to release the information to WS **166**. HS **116** then prepares a response including a user identifier, such as a GUPI, the requested information, and a nonce provided with request. The response is signed by HS **116**, and a root-signed copy of HS **116**'s public signature is appended to the signed response. This response is forwarded to MS **118** via browser **114** by redirecting the browser, along the continuation of datapath **176**, using any of a number of known techniques. MS **118** then forwards the segment of the signed response corresponding to the request from WS **166** to WS **166** over datapath **178**. After verifying the nonce and the requested information, WS **166** obtains the flight information for the user, provides it to MS **118**, and allows any of a number of functions to be provided including seat selection and advance check-in with electronic boarding pass provisions. One skilled in the art will appreciate both that other services can be provided, and that MS **118** can connect to a plurality of webservices to aggregate data from each of them. In one embodiment of the present invention, MS **118** requests sessions with a plurality of webservice providers, and aggregates their information requests. The aggregated requests are then provided en masse to HS **116**, and user authorization for all requests is obtained at once. This allows HS **116** to provide a series of responses to MS **118**, at which time MS **118** then separates the responses and sends each of the individual responses to the respective webserivce providers. The severing of the concatenated responses from HS **116** can easily be managed using the session identifiers issued by each webservice provider as a key. In alternate embodiments, HS **116** obtains user approval for the release of the information to each of the

webservice providers, and then sends the responses one at a time to MS **118**, which after receiving a response simply redirects browser **114** to HS **116** to obtain the next response until all responses are obtained and forwarded to the webservice providers. One skilled in the art will appreciate that the actual mechanism used for the supporting of multiple webservice providers can vary without affecting the scope of the present invention.

Webservice providers, such as WS **166**, can relate the information that they requested to their database, and at the same time be assured that they are allowed to release the information, as HS **116** can be proven to be authoritative for the user's identifier by following a signature key chain through any number of delegations until a trusted source is used to show that HS **116** is authoritative for the information released.

Some of the information housed by HS **116** may be provided to it by an outside authoritative source as described in detail in related applications, such as Canadian Application Numbers 2,468,351, and 2,468,585, which are hereby incorporated by reference. In cases where the information, such as a frequent flier number, is housed by an external authoritative site, HS **116** can provide the externally signed assertion to WS **166**, allowing WS **166** to determine both that the information provided is authentic, and that HS **116** is authoritative for the user associated with the information.

One skilled in the art will appreciate that MS **118** may always connect to WS **166**. As a result, MS **118** can, upon receiving an indication that the user is part of the identity management system, initiate the connection to WS **166** to request a session over datapath **170**. When MS **118** requests information required by WS **166**, it can include its own user authentication request. Thus, authentication of the user at HS **116** can be done at the same time that the user authorizes the release of information to WS **166**.

As illustrated in Figure 10, HS **116** is authoritative for the user of browser **114**. As such, HS **116** can be used as an agent of the user and permitted to directly interact with WS **166**. In such a scenario HS **116** directly connects to WS **166**, and requests information using connection **182**. WS **166** uses a standard interface, and as a result does not notice a difference between HS **116** and MS **118**. Because HS **116** is already authoritative for the user, the authentication and data passing through MS **118** can be bypassed. WS **166** can issue an authentication request **184**, which HS **116**, acting as an agent for the user, can directly respond to over connection **186**. WS **166** can then either issue a token or one-time-

information **188**. The user's trust of HS **116** allows for this scenario to be permitted, as without user approval HS **116** will not interact with WS **166**.

A rich client can be provided that interacts with WS **166** on the user's behalf without having to interact with MS **118**, as illustrated in Figure 11. As an example, if WS **166** provides financial information to users for a bank, the bank can provide users with a rich client (RC) **190** that will interact with WS120. The client is preferably a non-browser based application that can make calls to WS **166**. Due to its standardized interface, WS **166** is agnostic as to who interacts with it. RC **190** appears as another MS to WS **166**. When RC **190** issues a request **192** to WS **166**, it receives a request for authentication and information **194**. RC then launches a browser **114** from the local computer, and uses the browser **114** to transmit WS **166**'s request to HS **116** over datapath **196**. The user interacts with HS **116** in the normal manner, and approves the release of the information. The response **198** is sent from HS **116** to the browser **114**, which provides the information to RC **190**. RC **190** can then close the browser window, and forward the requested information to WS **166** over datapath **200**. RC **190** can obtain the requested information from the browser **114** prior to closing the window, by having the browser **114** redirected to the localhost address at a predetermined port. As long as RC **190** has control over that port and is listening on it, the information can be received and then sent along to WS **166**. WS **166** can then send the requested information, or a token, to RC **190** over datapath **202**.

To facilitate the interaction of RC with the rest of the network, RC can use a public API to interact with network nodes such as HS **116** and WS **166**. As new standards for connection are defined, or new node types arise, the API can be changed by a public administrator, and then provided to the users of RC. By replacing the API, the ability to connect either to new node types or to existing nodes in a new manner can be provided without requiring the rewriting of the code base for RC.

As described above, the network root administers admission to the network, and provides signed assertions that a homesite is authoritative for a user. Each user is uniquely identified by both a GUPI and an email address. By leveraging the trust model of this network, a distributed contact management network is provided. At present contact management networks require a single database of contacts that are maintained by a sole provider. The distributed nature of the network of the present invention bypasses the

drawbacks to that model. A distributed contact management system in the network of the present invention is illustrated in Figure 12.

Root **164** maintains a database **204** mapping email addresses to associated GUPI's and homesite identifiers used to identify the homesite that is authoritative for the GUPI. HS **116** is authoritative for a GUPI associated with the user of browser **114**. The user of browser **114** provides to HS **116** a listing of known contacts over datapath **206**. HS **116** extracts the email addresses from the contact listing and provides the email addresses to root **164** over datapath **208**. Root **164** then identifies the GUPI and homesite associated with each submitted email address, and provides this information to the HS **116** over return datapath **210**. HS **116** can then contact HS2 **212**, which is authoritative for a GUPI associated with one of the submitted email addresses over connection **214**. When HS **116** contacts HS2 **212** over datapath **214**, it can request additional contact information stored by HS2 **212**. HS2 **212** can release this information, if authorized by the relevant user, or can ask the relevant user for authorization at the next login. When providing the information, HS2 **212** can provide a URI to HS **116** allowing HS **116** to obtain updated information at other times, so that the contact information can be updated periodically. Conversely, HS **116** can provide HS2 **212** with a URI so that when the requested information changes, or at fixed intervals, HS **116** will receive updated information over datapath **214**. After receiving the user information over datapath **214**, HS **116** can forward the information to browser **114** over datapath **216**. One skilled in the art will appreciate that a number of other software applications, other than a standard internet web browser, can be used by the user to communicate with HS **116** including email and contact management clients. In the above scenario the contact information can be transmitted in any of a number of formats including the virtual card (vcard) standard.

The above-described scenario allows homesites to communicate to each other using URI's to update information. A similar network service is illustrated in Figure 13, where HS **116** is provided with an update URI by MS **118**, through browser **114** during a request for information over datapath **218**. The requested information is provided to MS **118** over datapath **220**. Both channels **218** and **220** make use of the redirection of browser **114**. However, when the supplied information changes, and if the user has approved the updating of information, HS **116** can create a back channel connection **222** to MS **118** to supply the

updated information. One skilled in the art will appreciate that in addition to the request interface described above, a homesite would preferably have an update interface or engine, to allow monitoring of information for a user, so that when a user modifies a profile, the relevant information can be updated by backchannel, without requiring user interaction with the membersite.

Figure 14 illustrates the use of the update URI for a user who is changing from HS **116** to HS2 **212** as a principal identity store. As in the example of Figure 13, HS **116** has obtained an update URI associated with MS **118** over datapath **218**, and has provided contact information to MS **118** over data path **220**. The user, through browser **114** over datapath **224**, informs HS **116** that the GUPI, and all associated information, is to be transferred to HS2 **212**. This can be a permanent transfer of information, or it can be using one homesite to serve as a backup to another. HS **116** connects to HS2 **212**, either directly over a back channel, or through redirection of the browser **114**, and an exchange of data is made. This datapath is shown as datapath **226**, which is an example of a back channel connection, but one skilled in the art will appreciate that it can be performed using browser redirection. Along with the user information associated with the GUPI, HS **116** transfers the update URI from MS **118** to HS2 **212**. Thus, when HS2 **212** receives updated information from browser **114** the information can still be updated with MS **118** seamlessly over datapath **228**.

GUPI's are typically assigned by root **164** to a homesite, such as HS **116**. Thus far in identity management systems, each identifier is linked to an email address. This removes the ability of a user to be anonymous, as the identifier can be associated with an email address that is easily traceable to a user. To satisfy the need for anonymous personas, root **164** can assign a series of GUPIs to HS **116** as an anonymous pool. This allows HS **116** to provide a user with a pool of anonymous GUPIs, so that if a user wishes to remain anonymous, HS **116** is the only site that can identify the user. Once again, this model is predicated upon the user of browser **114** having trust in HS **116**, without which, HS **116** would never be able to server as a homesite that stores the user's identity information. With a sufficiently large pool of anonymous GUPIs, HS **116** can assign a different GUPI to each site that a user visits. Though this prevents the building of attributes that can establish a virtual reputation, the purpose of anonymous personas is to prevent the building of any reputation. Because no two

sites will be given the same GUPI, the result is much the same as a pairwise unique identifier, however, HS **116** can, in one embodiment, economize on GUPI's in the unique pool by allowing the same GUPI to be used by two different users at two different sites. Because the GUPI has no attributes associated with it, and no user can build a reputation with it, if treated communally it further anonymizes the behaviour of the user. In non-shared embodiments, HS **116** must track the pairings of the membersite identifier and the user to determine the GUPI to be used. If the GUPI is not shared, it is still globally unique, and can be ported to another homesite. When transferring persona information to another homesite, the user can obtain a GUPI list from the homesite and can have the authoritativeness of that GUPI transferred to another homesite. For the embodiment where GUPIs are communally shared, the new homesite can be made authoritative for the GUPI, maintaining the same membersite identifier and user pairing to associate to the GUPI, without revoking the authoritativeness of the original homesite, as other people at other sites may use the GUPI.

One of the issues that arise from using multiple GUPI to allow a user to keep persona separate, is that when assertions are made, they are typically made for a single persona. Thus, for a user with home and office persona, an assertion may be made for the office persona regarding membership in an organization. If the user's home persona needs to make use of the membership assertion attached to the office persona there are two mechanisms provided by the present invention for this. Using a first mechanism, a user can direct HS **116** to contact AS which issued the assertion for the work persona. HS **116** then provides AS with multiple GUPIs, and the assertions for any of the provided GUPIs issued by AS and indicates that it is authoritative for all the submitted GUPIs, and all the submitted GUPIs are the same individual. AS, upon being informed that all the GUPIs are issued to the same individual, can then provide any GUPIs that do not have an assertion, with the assertion provided by HS **116**. In an example, AS is a frequent flier program, and has provided an assertion indicating that the office persona of a user has obtained elite status. HS **116** provides the GUPIs for the user's office and home persona to AS along with the assertion that the office persona has obtained elite status. AS then verifies that HS **116** is authoritative for both GUPIs, and confirms that the office persona is certified as having elite status. AS then provides an assertion for the GUPI associated with the home persona

indicating elite status. This allows for assertions to be shared between persona, but comes at the cost of having AS know that two GUPI are related to each other.

If a user wishes to avoid having two GUPI linked together by an AS, but one GUPI has an assertion needed by the other GUPI, the following method can be employed. When MS **118** requests an assertion about a first GUPI that is only held by a second GUPI, HS **116** can include in its signed response, both GUPIs, and the assertion held for the second GUPI. MS **118** can then determine from the response, that HS **116** is authoritative for both GUPIs, and sees that HS **116** states that both GUPIs are issued to the same person. MS **118** can then verify that the second persona has the required assertion, and apply the assertion to the first persona. As an example, a user with office and home persona has an assertion for the office persona that indicates elite status in a frequent flier program from AS. The user wants to use this assertion with the home persona when visiting MS **118**. In response to MS **118**'s request for the assertion, HS **116** sends both office and home GUPI, and the assertion for the office persona GUPI. MS **116** can then verify that the GUPIs are related, and can transitively apply the assertion to the home persona. In contrast to the first embodiment, AS does not know that the persona are linked, but MS **118** knows. Because the operation, by default, includes moving attributes from one persona to another, HS **116** must reveal the link between 2 GUPIs to at least one of the two. By offering both mechanisms, the user is provided the opportunity to choose which node in the network is shown the link.

The above described method and system for sharing credentials between GUPIs can also be used in relation to anonymous GUPI, though it should be noted that this reduces the anonymity of a GUPI, so should preferably not be done with a GUPI shared among users. For a MS **118** that has only ever been presented with an anonymous GUPI, the above-described method provides a method of transferring history to an identifiable GUPI. If HS **116** provides MS **118** with both an anonymous GUPI and an identifiable, or non-anonymous, GUPI, MS **118** can transfer any history associated with the anonymous GUPI to the identifiable GUPI. This allows a user to interact with MS **118** in an anonymous fashion, and then, having reached a comfort level with MS **118**, the user can present another GUPI and have any history and reputation transferred to the non-anonymous GUPI.

To increase the availability of homesite management capabilities, a homesite can be built-in to a browser. Such a homesite can be offered either as an integral part of a web

browser, or can be offered using a plug-in architecture. Such a plug in, or integrated browser, can be used to simplify the communication with nodes in the network and reduce the redirection of previous embodiments.

At a first level, a browser can indicate that it understands extensions to HTML specific to the identity management network. When browsers make requests from web servers using the hypertext transfer protocol (http), they provide an indication of capabilities, including an HTML version. By indicating that the browser understands the identity management network extensions to HTML (or identity management HTML tags), MS **118** does not need to redirect the browser to the shadow domain to find out the homesite of the user. Instead, MS can simply send an HTML instruction to the browser to obtain user authentication. If the browser indicates that it is both identity management aware, and that a homesite has been configured, MS **118** need only provide authentication and information requests to the browser, and the browser will then handle any redirection needed. This allows MS **118** to avoid using redirections to shadow domains to find out what the user's homesite is, and avoids having to issue redirection requests to the browser. From the perspective of the user, fewer redirection requests are issued, and MS **118** never obtains the location of the users' homesite. If MS **118** simply instructs the enhanced browser to obtain authentication in an HTML tagged message, it does not need to tell the browser where to redirect to, and avoids using javascript™ redirects and close window commands to make the user experience seamless. The MS only determines the user's HS, when a response is issued, which increases user privacy.

As an enhancement, an enhanced browser can also be provided with the ability to function as a homesite. As disclosed in the above-cited references, a homesite can be provided as a local application. By integrating the homesite within the web browser, redirection can be avoided. When the HS-enabled browser visits MS **118**, it indicates that it supports identity management HTML tags. MS **118** then instructs the browser to obtain user authentication and return user information. HS-enabled browser no longer needs to redirect to an external site, and instead can provide user authentication using a locally controlled authentication tab or window. If the user has specified that use of the browser is a sufficient indication of authentication, HS-enabled browser can immediately return the requested information, having signed the response. This eliminates the user having to interact with an

external homesite, and reduces the data transmission, which is especially important on low-bandwidth connections. The HS-enabled browser preferably does not have a homesite cookie, so that MS **118** will not know that the user is using a local homesite.

One skilled in the art will appreciate that when an identity management aware browser sends identity management information through http headers, it allows MS **118** to refrain from bouncing the browser to the shadow domain. This allows MS **118** to simplify its interaction with the browser, as the browser has indicated that it knows a homesite for the user. Instead of the MS being sent the HS identifying information, MS uses an http command to request authentication in a POST command. The browser will handle redirection if needed and will replace the request authentication command with the appropriate HTML if an external HS is used. If an external HS is used, it can identify that it does not need to use a redirect command to send the information to MS, and instead simply sends the response to the browser and tells the browser to send the information to the MS.

The above-described enhancement to a browser can either be integrated into the browser code, or can be provided as a plug in. One skilled in the art will appreciate that either embodiment can communicate with a root node to obtain updated schema, or can obtain the updated schema from a central service used to ensure that the browser has been updated to the most recent patches and bug-fixes.

One skilled in the art will appreciate that the homesites, webservices and membersites of the present invention can all be implemented on standard computing hardware using known software techniques to implement the methods of the present invention. These systems typically include an input to receive requests from external nodes, a processor for examining the request and for acting upon the request in accordance with the methods of the present invention, and an output for issuing both requests and responses as needed or required by the methods of the present invention. The implementation of the methods of the present invention on such hardware, using either conventional software or firmware, are well within the scope of one of skill in the art.

The above-described embodiments of the present invention are intended to be examples only. Alterations, modifications and variations may be effected to the particular embodiments by those of skill in the art without departing from the scope of the invention, which is defined solely by the claims appended hereto.

CLAIMS:

1. A method of obtaining contact information from a node in an identity management network, the method comprising:

receiving a homesite identifier and a user identifier associated with an email address;

requesting from a homesite associated with the received homesite identifier contact information for a user associated with the received user identifier; and

receiving from the homesite associated with the received homesite identifier the requested contact information.

2. The method of claim 1 further including the step of forwarding the received contact information to a user.

3. The method of claim 1 wherein the step of receiving a homesite identifier includes receiving the homesite identifier and user identifier from a network root.

4. The method of claim 3 wherein the step of receiving a homesite identifier is preceded by the step of transmitting a request for a user identifier and a homesite identifier associated with an email address.

5. The method of claim 1 wherein the user identifier is a globally unique persona identifier.

6. The method of claim 1 wherein the step of receiving the requested contact information is preceded by obtaining approval for the release of the contact information from the user associated with the requested contact information.

7. The method of claim 1 wherein the step of receiving the requested contact information includes receiving a universal resource indicator for receiving updated contact information.

8. A method of authorizing a webservice provider to release information, associated with a user, to a third party, the method comprising:

receiving from the webservice provider a request for user authentication and authorization to release information to the third party;

authenticating the user and obtaining user authorization for the release of the information; and

forwarding to the webservice provider authorization from the authenticated user to release the information to the third party.

9. The method of claim 8 wherein the received request from the webservice provider is received via the third party.

10. The method of claim 8 wherein the received request includes an explanation of the information to be released to the third party.

11. The method of claim 10 wherein the explanation is a text explanation to be shown to the user prior to obtaining user authorization.

12. The method of claim 10 wherein the explanation is a programmatic explanation.

13. The method of claim 8 wherein the received request includes a nonce.

14. The method of claim 13 wherein the step of forwarding includes forwarding the nonce along with the response.

15. The method of claim 8 wherein the step of forwarding includes signing the response.

16. The method of claim 8 wherein the step of forwarding includes appending proof of authoritativeness for the user.

17. The method of claim 8 wherein the step of forwarding includes forwarding the authorization to the webservice provide via the third party.

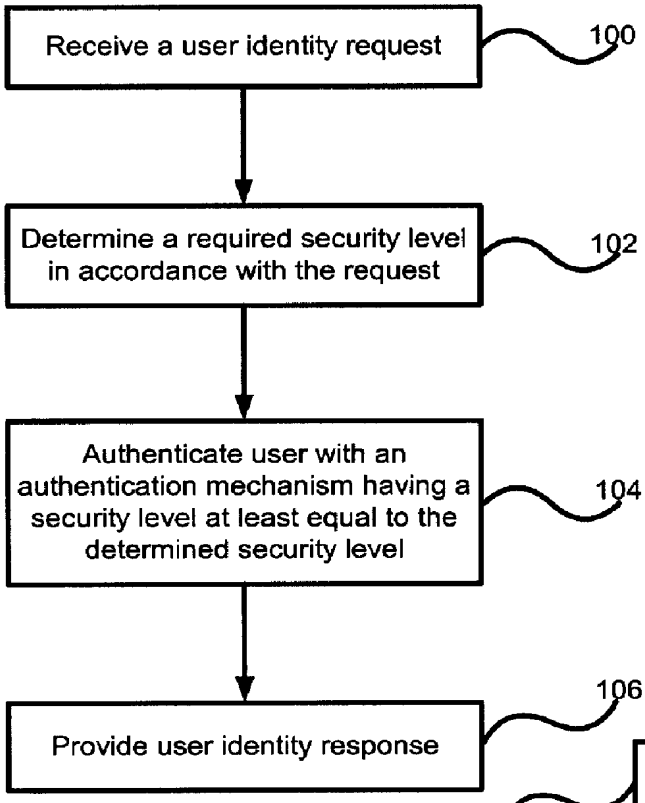


Figure 1

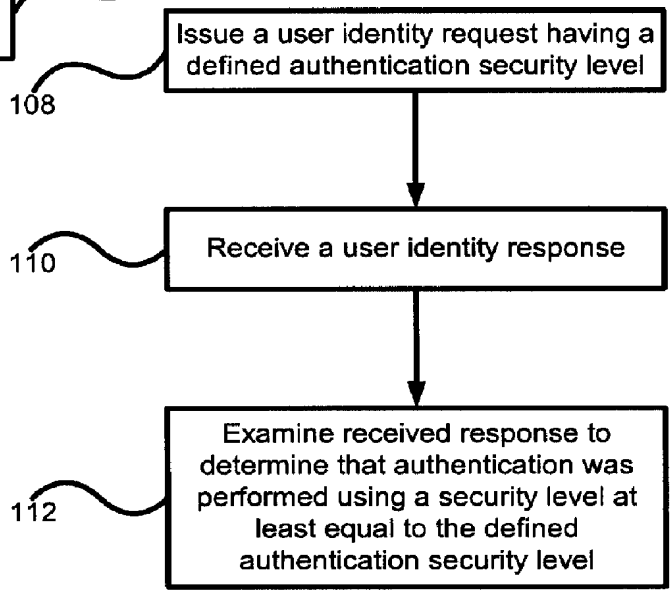


Figure 2

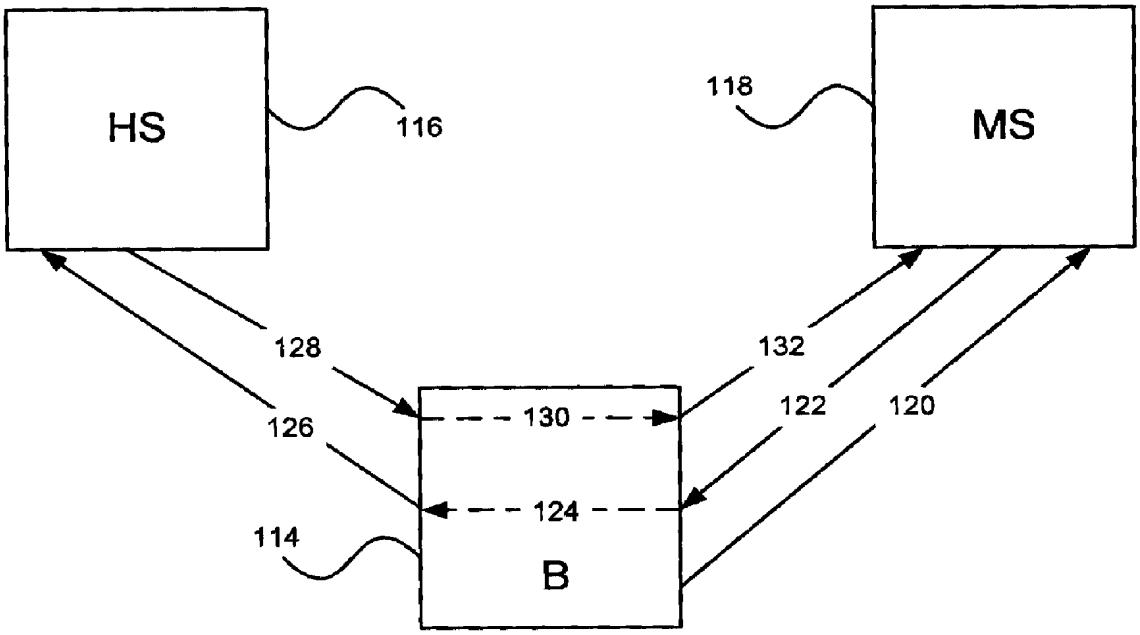


Figure 3

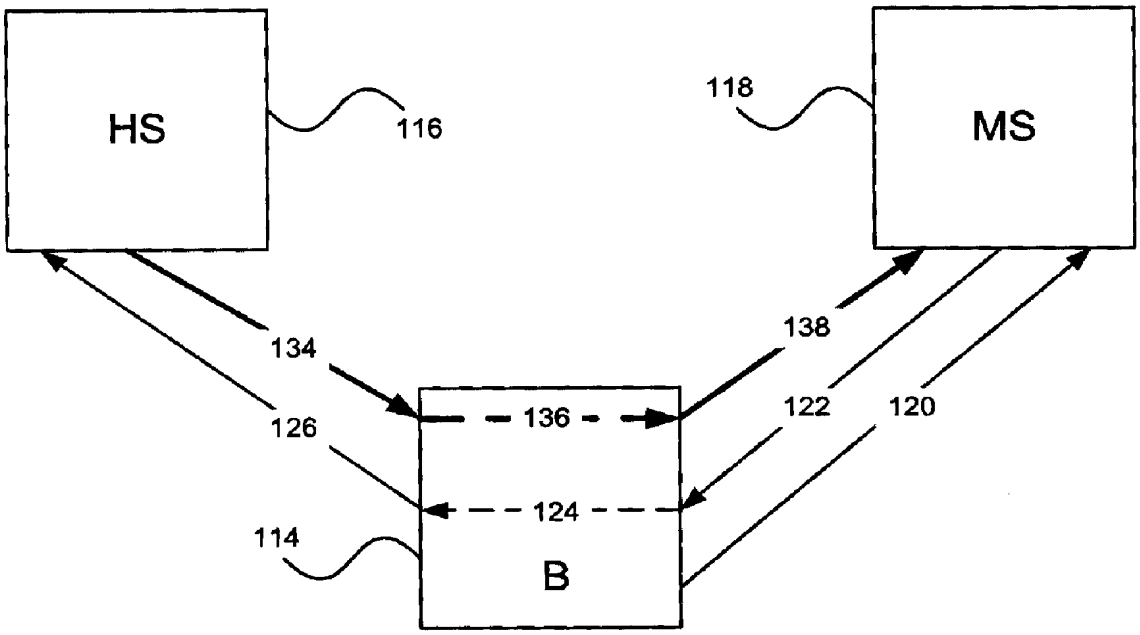


Figure 4

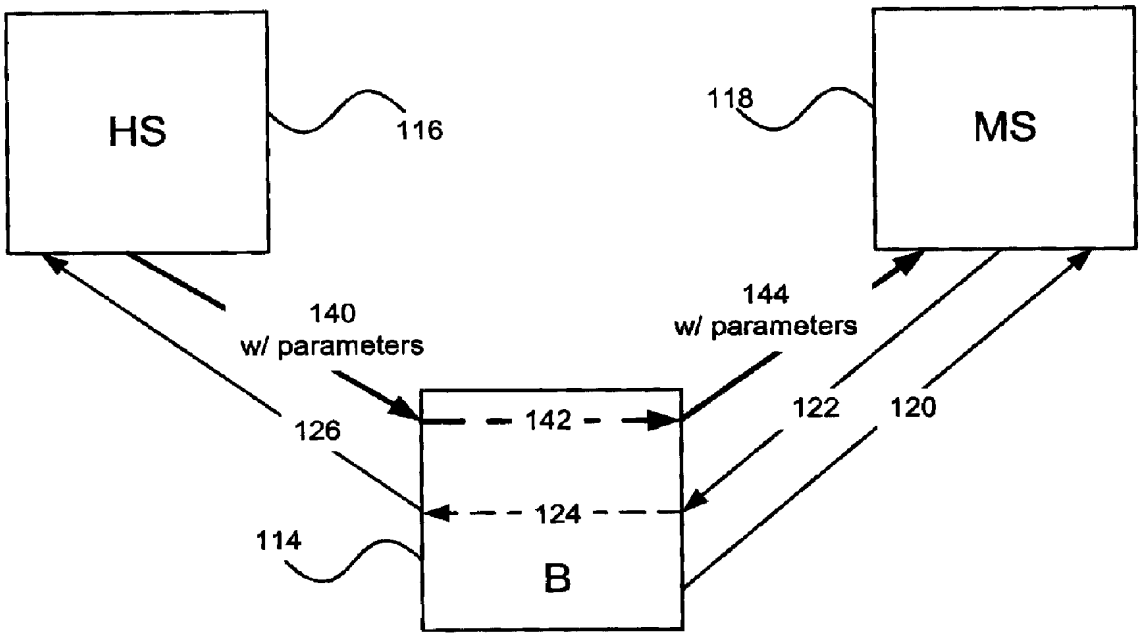


Figure 5

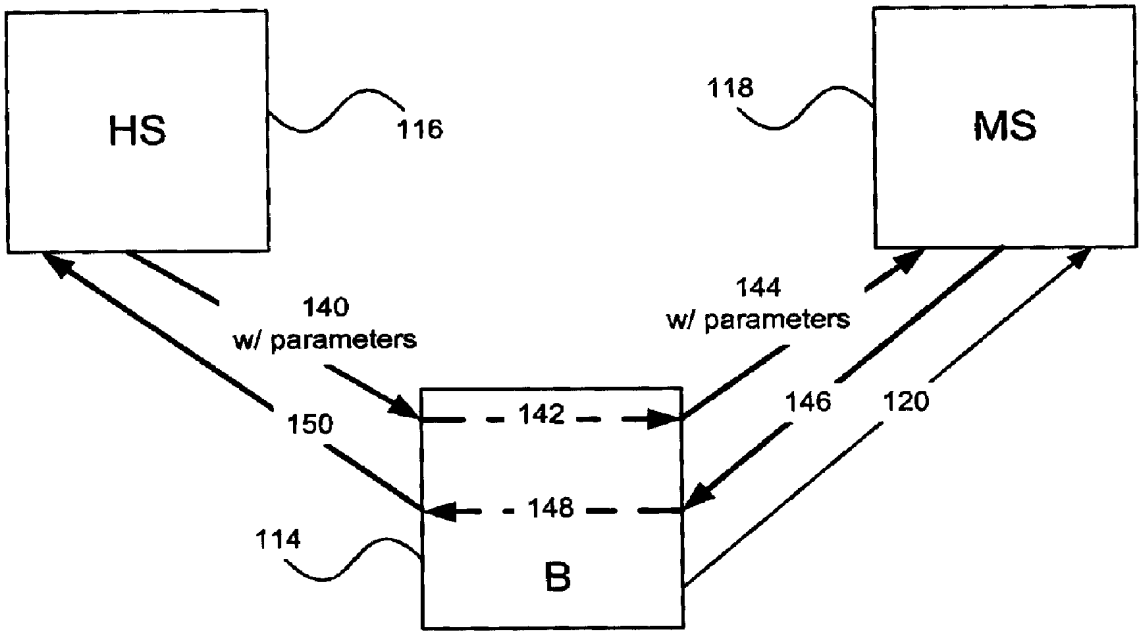


Figure 6

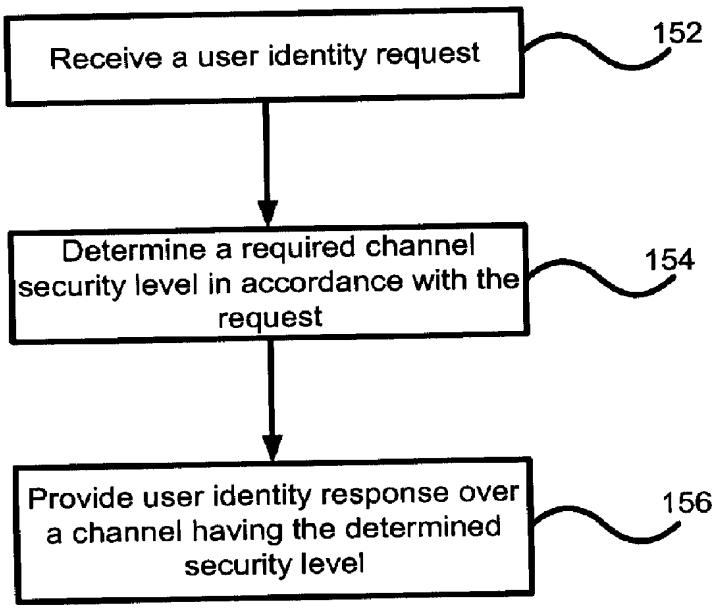


Figure 7

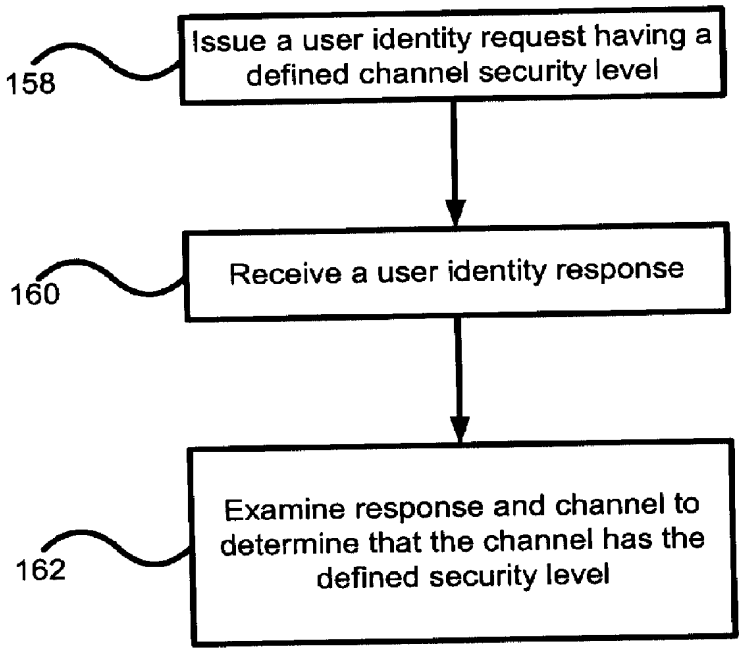


Figure 8

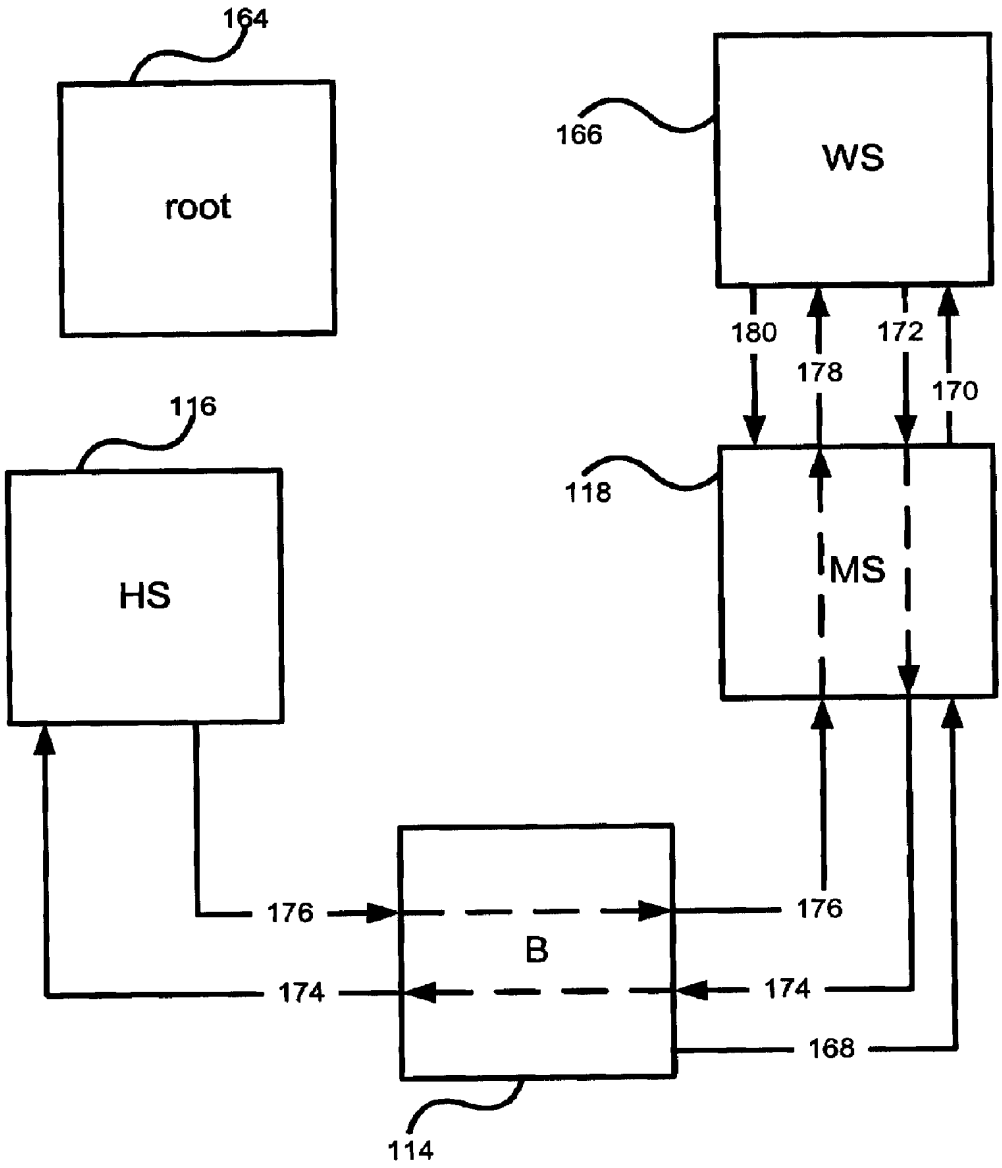

164
root
166
WS
180
178
172
170
116
HS
118
MS
176
176
B
174
174
168
114

Figure 9

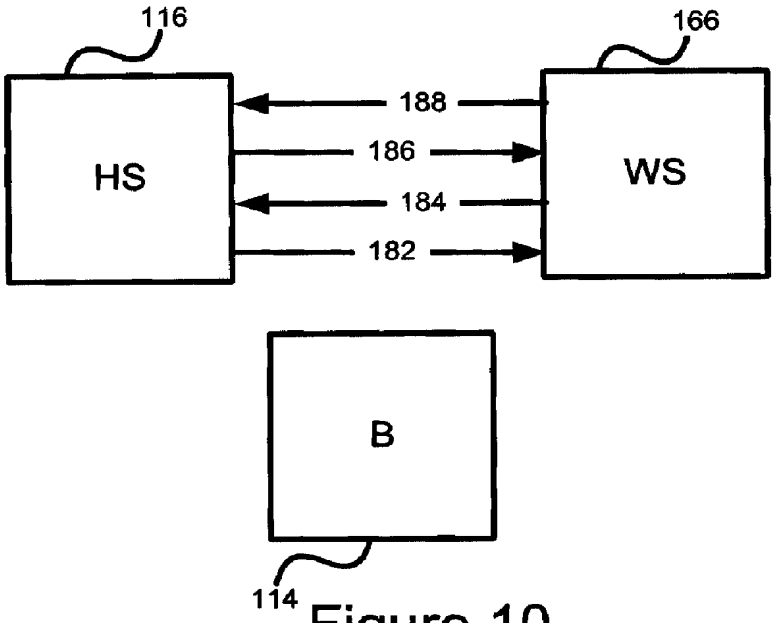


Figure 10

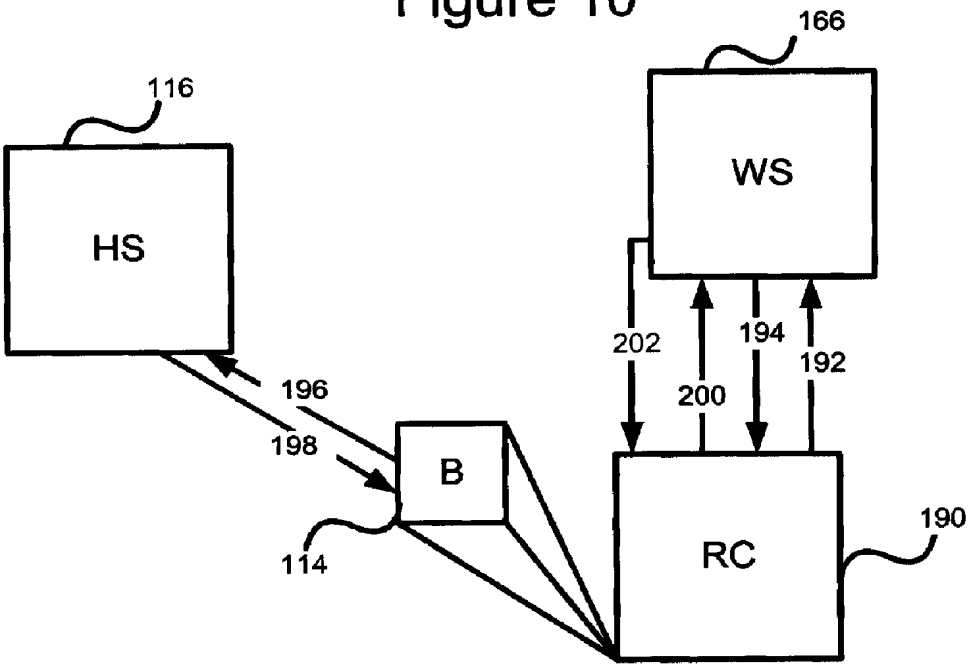


Figure 11

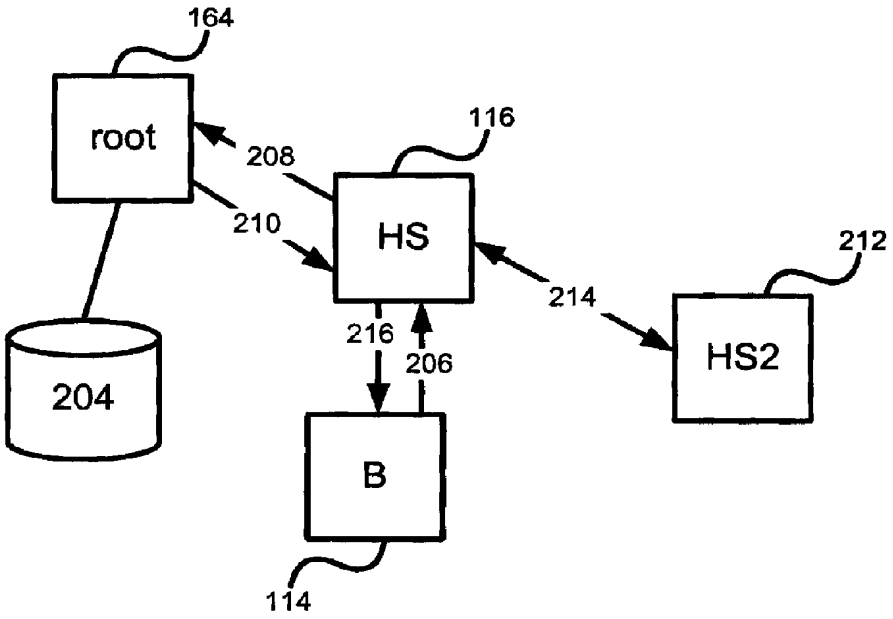


Figure 12

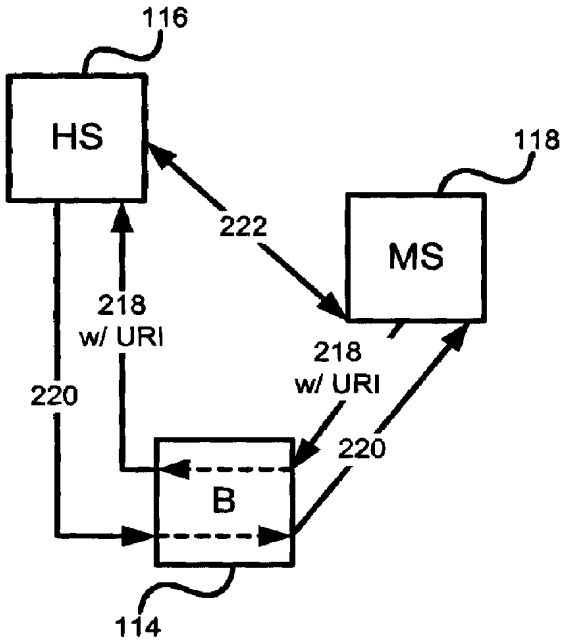


Figure 13

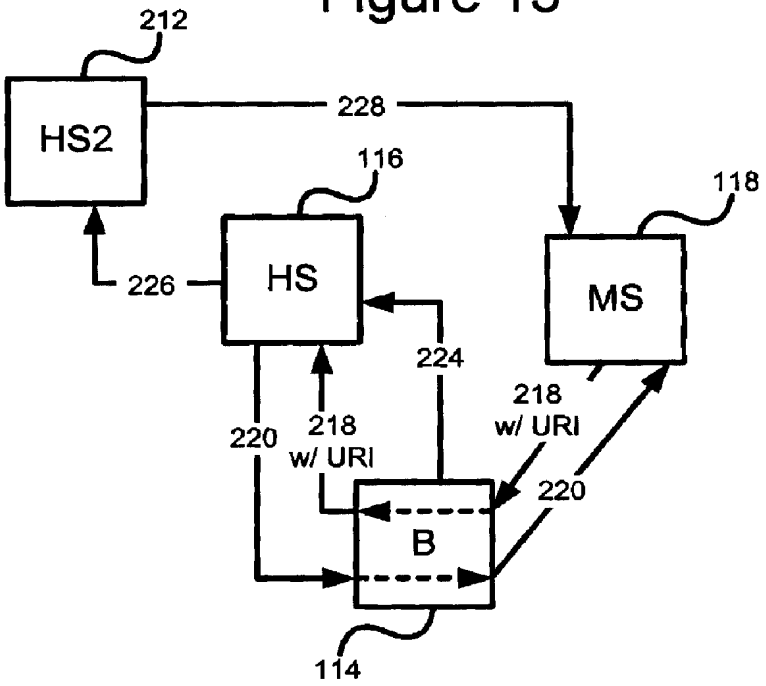


Figure 14